**Examen de Programación C# ASP.NET**

Jennifer Rebeca García Moncada

Universidad Politécnica Internacional

Programación II

Profesor: Alexander Benjamin Curling

Julio, 2024

**Análisis del Escenario de Recolección de Datos y Diseño de la Base de Datos**

**Objetivos del sistema:**

* Permite a los votantes registrarse y votar por los candidatos.
* Registrar y almacenar información de partidos, candidatos, votantes y votos.
* Calcular y mostrar los resultados de las elecciones.

**Entidades Identificadas:**

**Partidos:** Información sobre los partidos políticos.

**Candidatos:** Información sobre los candidatos de los partidos.

**Votantes:** Información sobre los votantes que pueden emitir votos.

**Votos:** Registro de los votos emitidos por los votantes.

**Resultados:** Información sobre los resultados de las elecciones.

**Requisitos Funcionales:**

Registro de partidos con nombre, siglas y fecha de fundación.

Registro de candidatos con cédula, nombre, apellidos, dirección, partido al que pertenecen y fecha de registro.

Registro de votantes con cédula, nombre, apellidos, fecha de nacimiento, email y dirección.

Registro de votos emitidos por los votantes, relacionando cada voto con un candidato.

Cálculo de los resultados de las elecciones mostrando la cantidad de votos y el porcentaje de votos por candidato.

**Requisitos No Funcionales:**

**Seguridad:** Asegurar la integridad y confidencialidad de la información de los votantes.

**Rendimiento:** Manejo eficiente de las consultas para la visualización de resultados en tiempo real.

**Usabilidad:** Interfaz amigable para el registro y visualización de información.

**Diseño de la Base de Datos**

**Tablas y Relaciones**

1. **Partidos:**

**Campos:** Id (PK), Nombre, Siglas, FechaFundacion

**Restricciones:** Nombre y Siglas deben tener longitud mayor a 0, Siglas deben ser únicas.

1. **Candidatos:**

**Campos:** Id (PK), Cedula, Nombre, Apellido1, Apellido2, Direccion, IDPartido (FK), FechaRegistro

**Restricciones:** Cedula debe seguir un formato específico, Nombre y IDPartido deben ser únicos en combinación, IDPartido debe existir en la tabla Partidos.

1. **Votantes:**

**Campos**: IDVotante (PK), Nombre, Apellido1, Apellido2, FechaNacimiento, Email, Direccion

**Restricciones:** IDVotante debe seguir un formato específico, Email debe ser único.

1. **Votos:**

**Campos:** Id (PK), IDCandidato (FK), IDVotante (FK), FechaVoto

**Restricciones:** IDCandidato debe existir en la tabla Candidatos, IDVotante debe existir en la tabla Votantes, un votante solo puede emitir un voto (IDVotante debe ser único), FechaVoto no puede ser futura.

1. **Resultados:**

**Campos:** Id (PK), IDCandidato (FK), CantidadVotos, Porcentaje, Fecha

**Restricciones:** IDCandidato debe existir en la tabla Candidatos, CantidadVotos debe ser mayor o igual a 0, Porcentaje debe estar entre 0 y 100.

**Código SQL de la Base de Datos**

**Texto

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

**Inserción de Datos de Prueba**

**Texto

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

**Texto

Descripción generada automáticamente**

**Texto

Descripción generada automáticamente**

**Texto

Descripción generada automáticamente**

**Visualización de los datos**

**Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente**

**Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media**

**Código Fuente de la aplicación WEB**

**Site.css**

/\* Estilo general del cuerpo de la pagina\*/

body {

font-family: Arial, sans-serif;

background-color: #e0f7fa; /\* Fondo azul claro \*/

}

/\* Estilos para el menú vertical\*/

.menu-vertical {

width: 200px;

float: left;

background-color: #ffcdd2; /\* Fondo rojo claro \*/

border-radius: 10px;

padding: 10px;

}

/\* Elimina los puntos de la lista del menú \*/

.menu-vertical ul {

list-style-type: none;

padding: 0;

}

/\* Estilos para los elementos de la lista del menú \*/

.menu-vertical ul li {

margin: 10px 0;

}

/\* Estilos para los enlaces del menú \*/

.menu-vertical ul li a {

text-decoration: none;

color: #000;

padding: 10px;

display: block;

background-color: #f4f4f4;

border: 1px solid #ddd;

border-radius: 4px;

}

/\* Cambia el fondo al pasar el cursor sobre los enlaces del menú \*/

.menu-vertical ul li a:hover {

background-color: #ddd;

}

/\* Estilo del contenido principal, incluye margen para dar espacio al menú \*/

#MainContent {

margin-left: 220px;

padding: 20px;

}

/\* Estilo para los contenedores principales \*/

.container {

max-width: 800px;

margin: 0 auto;

padding: 20px;

border: 1px solid #ccc;

border-radius: 10px;

background-color: #f9f9f9;

}

/\* Estilo para formularios \*/

.form-container {

width: 100%;

margin: 0 auto;

}

/\* Estilo para los grupos de formularios\*/

.form-group {

margin-bottom: 15px;

}

/\*Estilo para las etiquetas de los formularios \*/

.form-group label {

display: block;

font-weight: bold;

margin-bottom: 5px;

}

/\*Estilo para los controles de los formularios\*/

.form-control {

width: 100%;

padding: 10px;

font-size: 14px;

box-sizing: border-box;

}

/\* Centrar texto \*/

.text-center {

text-align: center;

}

/\* Estilo pra los botones \*/

.btn {

display: inline-block;

padding: 10px 20px;

font-size: 16px;

cursor: pointer;

text-align: center;

text-decoration: none;

outline: none;

color: #fff;

background-color: #007bff;

border: none;

border-radius: 5px;

}

/\* Cambia el color de fondo de los botones al pasar el cursor \*/

.btn:hover {

background-color: #0056b3;

}

/\* Cambia el color de fondo de los botones al hacer click \*/

.btn:active {

background-color: #003d82;

transform: translateY(2px);

}

/\* Estilo para la tabla de resultados \*/

.tabla-resultados {

width: 100%;

border-collapse: collapse;

margin: 20px 0;

font-size: 18px;

text-align: left;

}

/\* Estilo para las celdad de la tabla de resultados \*/

.tabla-resultados th, .tabla-resultados td {

padding: 12px;

border-bottom: 1px solid #ddd;

}

/\* Estilo para las cabeceras de la tabla de resultados \*/

.tabla-resultados th {

background-color: #4CAF50;

color: white;

}

/\* Color de fondo para las filas pares de la tabla de resultados \*/

.tabla-resultados tr:nth-child(even) {

background-color: #f2f2f2;

}

/\* Cambia el color de fondo al pasar el cursor sobre las filas de la tabla \*/

.tabla-resultados tr:hover {

background-color: #ddd;

}

/\* Estilo especial para resaltar al ganador \*/

.ganador {

background-color: #ffeb3b !important; /\* Amarillo \*/

}

/\* Estilo para centrar la tabla \*/

.table-container {

display: flex;

justify-content: center;

align-items: center;

}

**Default.aspx**

<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true" CodeBehind="Default.aspx.cs" Inherits="Sistema\_Votaciones.Default" %>

<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">

<title>Inicio</title>

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">

<div class="text-center">

<h2>Bienvenido al Sistema de Votaciones</h2>

<img src="Content/Bienvenido.jpg" alt="Bienvenido" style="width:100%; max-width:600px;" />

</div>

</asp:Content>

**IngresarCandidatos.aspx**

<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true" CodeBehind="IngresarCandidatos.aspx.cs" Inherits="Sistema\_Votaciones.IngresarCandidatos" %>

<!-- Contenido para la sección de encabezado -->

<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">

<!-- Título de la página -->

<title>Ingresar Candidatos</title>

<!-- Importar jQuery -->

<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>

<!-- Importar Inputmask para formateo de campos de entrada -->

<script src="https://cdnjs.cloudflare.com/ajax/libs/inputmask/5.0.6/inputmask.min.js"></script>

<!-- Script para formatear el campo de fecha de nacimiento -->

<script type="text/javascript">

$(document).ready(function () {

var selector = $("#<%= txtFechaNacimiento.ClientID %>");

Inputmask("9999-99-99", { alias: "yyyy-mm-dd", placeholder: "YYYY-MM-DD" }).mask(selector);

Inputmask("99-99-9999", { alias: "dd-mm-yyyy", placeholder: "DD-MM-YYYY" }).mask(selector);

});

</script>

</asp:Content>

<!-- Contenido principal de la página -->

<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">

<div class="container">

<!-- Título centrado de la sección -->

<h2 class="text-center">Ingresar Candidatos</h2>

<div class="form-container">

<!-- Grupo de formulario para la cédula -->

<div class="form-group">

<label for="txtCedula">Cédula:</label>

<asp:TextBox ID="txtCedula" runat="server" CssClass="form-control" placeholder="Ingrese la Cédula"></asp:TextBox>

</div>

<!-- Grupo de formulario para el nombre -->

<div class="form-group">

<label for="txtNombre">Nombre:</label>

<asp:TextBox ID="txtNombre" runat="server" CssClass="form-control" placeholder="Ingrese el Nombre"></asp:TextBox>

</div>

<!-- Grupo de formulario para el primer apellido -->

<div class="form-group">

<label for="txtApellido1">Primer Apellido:</label>

<asp:TextBox ID="txtApellido1" runat="server" CssClass="form-control" placeholder="Ingrese el Primer Apellido"></asp:TextBox>

</div>

<!-- Grupo de formulario para el segundo apellido -->

<div class="form-group">

<label for="txtApellido2">Segundo Apellido:</label>

<asp:TextBox ID="txtApellido2" runat="server" CssClass="form-control" placeholder="Ingrese el Segundo Apellido"></asp:TextBox>

</div>

<!-- Grupo de formulario para la dirección -->

<div class="form-group">

<label for="txtDireccion">Dirección:</label>

<asp:TextBox ID="txtDireccion" runat="server" CssClass="form-control" placeholder="Ingrese la Dirección"></asp:TextBox>

</div>

<!-- Grupo de formulario para la fecha de nacimiento -->

<div class="form-group">

<label for="txtFechaNacimiento">Fecha de Nacimiento:</label>

<asp:TextBox ID="txtFechaNacimiento" runat="server" CssClass="form-control" placeholder="YYYY-MM-DD or DD-MM-YYYY"></asp:TextBox>

<asp:CustomValidator ID="cvFechaNacimiento" runat="server" ControlToValidate="txtFechaNacimiento"

ErrorMessage="Fecha inválida. Use el formato YYYY-MM-DD o DD-MM-YYYY."

OnServerValidate="cvFechaNacimiento\_ServerValidate"></asp:CustomValidator>

</div>

<!-- Grupo de formulario para seleccionar el partido -->

<div class="form-group">

<label for="ddlPartidos">Partido:</label>

<asp:DropDownList ID="ddlPartidos" runat="server" CssClass="form-control"></asp:DropDownList>

</div>

<!-- Botones para ingresar y consultar candidatos -->

<div class="text-center">

<asp:Button ID="btnSubmit" runat="server" CssClass="btn" Text="Ingresar Candidato" OnClick="btnSubmit\_Click" />

<asp:Button ID="btnConsultar" runat="server" CssClass="btn" Text="Consultar Candidatos" OnClick="btnConsultar\_Click" />

</div>

</div>

<!-- Tabla para mostrar los candidatos ingresados -->

<asp:GridView ID="gvCandidatos" runat="server" AutoGenerateColumns="False" CssClass="table">

<Columns>

<asp:BoundField DataField="Cedula" HeaderText="Cédula" />

<asp:BoundField DataField="Nombre" HeaderText="Nombre" />

<asp:BoundField DataField="Apellido1" HeaderText="Apellido1" />

<asp:BoundField DataField="Apellido2" HeaderText="Apellido2" />

<asp:BoundField DataField="Direccion" HeaderText="Dirección" />

<asp:BoundField DataField="Partido" HeaderText="Partido" />

<asp:BoundField DataField="FechaRegistro" HeaderText="Fecha de Registro" />

</Columns>

</asp:GridView>

</div>

</asp:Content>

**IngresarCandidatos.aspx.cs**

using System;

using System.Data.SqlClient; // Espacio de nombres para trabajar con SQL Server

using System.Web.Configuration; // Espacio de nombres para trabajar con la configuración de la aplicación web

using System.Data; // Espacio de nombres para trabajar con datos

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Configuration; // Espacio de nombres para trabajar con la configuración de la aplicación

namespace Sistema\_Votaciones

{

public partial class IngresarCandidatos : System.Web.UI.Page

{

// Método que se ejecuta cuando la página se carga

protected void Page\_Load(object sender, EventArgs e)

{

// Verifica si la página se está cargando por primera vez

if (!IsPostBack)

{

// Cargar los partidos en el dropdown

CargarPartidos();

}

}

// Método que se ejecuta cuando se hace clic en el botón "Ingresar Candidato"

protected void btnSubmit\_Click(object sender, EventArgs e)

{

// Verifica si la página es válida

if (Page.IsValid)

{

// Obtiene los valores de los campos de entrada

string cedula = txtCedula.Text;

string nombre = txtNombre.Text;

string apellido1 = txtApellido1.Text;

string apellido2 = txtApellido2.Text;

string direccion = txtDireccion.Text;

int idPartido = int.Parse(ddlPartidos.SelectedValue);

// Obtiene la cadena de conexión desde el archivo de configuración

string connectionString = ConfigurationManager.ConnectionStrings["BDVotacionesConnectionString"].ConnectionString;

using (SqlConnection con = new SqlConnection(connectionString))

{

// Consulta SQL para insertar un nuevo candidato

string query = "INSERT INTO Candidatos (Cedula, Nombre, Apellido1, Apellido2, Direccion, IDPartido) VALUES (@Cedula, @Nombre, @Apellido1, @Apellido2, @Direccion, @IDPartido)";

using (SqlCommand cmd = new SqlCommand(query, con))

{

// Agrega los parámetros a la consulta SQL

cmd.Parameters.AddWithValue("@Cedula", cedula);

cmd.Parameters.AddWithValue("@Nombre", nombre);

cmd.Parameters.AddWithValue("@Apellido1", apellido1);

cmd.Parameters.AddWithValue("@Apellido2", apellido2);

cmd.Parameters.AddWithValue("@Direccion", direccion);

cmd.Parameters.AddWithValue("@IDPartido", idPartido);

con.Open(); // Abre la conexión a la base de datos

cmd.ExecuteNonQuery(); // Ejecuta la consulta SQL

}

}

LimpiarCampos(); // Limpia los campos de entrada

}

}

// Método para validar la fecha de nacimiento

protected void cvFechaNacimiento\_ServerValidate(object source, ServerValidateEventArgs args)

{

DateTime fechaNacimiento;

string[] formatos = { "yyyy-MM-dd", "dd-MM-yyyy" }; // Formatos de fecha permitidos

// Intenta convertir el valor de entrada a una fecha

if (DateTime.TryParseExact(args.Value, formatos, System.Globalization.CultureInfo.InvariantCulture, System.Globalization.DateTimeStyles.None, out fechaNacimiento))

{

// Verifica si el candidato tiene al menos 18 años

if (DateTime.Now.Year - fechaNacimiento.Year >= 18)

{

args.IsValid = true; // La fecha es válida

}

else

{

args.IsValid = false; // La fecha no es válida

}

}

else

{

args.IsValid = false; // La fecha no es válida

}

}

// Método para cargar los partidos en el dropdown

private void CargarPartidos()

{

// Obtiene la cadena de conexión desde el archivo de configuración

string connectionString = ConfigurationManager.ConnectionStrings["BDVotacionesConnectionString"].ConnectionString;

using (SqlConnection con = new SqlConnection(connectionString))

{

// Consulta SQL para obtener los partidos

string query = "SELECT Id, Nombre FROM Partidos";

using (SqlCommand cmd = new SqlCommand(query, con))

{

con.Open(); // Abre la conexión a la base de datos

SqlDataReader reader = cmd.ExecuteReader(); // Ejecuta la consulta y obtiene un DataReader

ddlPartidos.DataSource = reader; // Establece la fuente de datos del dropdown

ddlPartidos.DataTextField = "Nombre"; // Campo que se mostrará en el dropdown

ddlPartidos.DataValueField = "Id"; // Campo que se usará como valor del dropdown

ddlPartidos.DataBind(); // Enlaza los datos al dropdown

}

}

// Agrega un elemento por defecto al dropdown

ddlPartidos.Items.Insert(0, new ListItem("Seleccione un partido", "0"));

}

// Método para limpiar los campos de entrada

private void LimpiarCampos()

{

txtCedula.Text = "";

txtNombre.Text = "";

txtApellido1.Text = "";

txtApellido2.Text = "";

txtDireccion.Text = "";

txtFechaNacimiento.Text = "";

ddlPartidos.SelectedIndex = 0;

}

// Método que se ejecuta cuando se hace clic en el botón "Consultar Candidatos"

protected void btnConsultar\_Click(object sender, EventArgs e)

{

// Obtiene la cadena de conexión desde el archivo de configuración

string connectionString = ConfigurationManager.ConnectionStrings["BDVotacionesConnectionString"].ConnectionString;

using (SqlConnection con = new SqlConnection(connectionString))

{

// Consulta SQL para obtener los candidatos y sus partidos

string query = @"

SELECT C.Cedula, C.Nombre, C.Apellido1, C.Apellido2, C.Direccion, P.Nombre AS Partido, C.FechaRegistro

FROM Candidatos C

JOIN Partidos P ON C.IDPartido = P.Id";

using (SqlCommand cmd = new SqlCommand(query, con))

{

con.Open(); // Abre la conexión a la base de datos

SqlDataReader reader = cmd.ExecuteReader(); // Ejecuta la consulta y obtiene un DataReader

gvCandidatos.DataSource = reader; // Establece la fuente de datos del GridView

gvCandidatos.DataBind(); // Enlaza los datos al GridView

}

}

}

}

}

**Resultados.aspx**

<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true" CodeBehind="Resultados.aspx.cs" Inherits="Sistema\_Votaciones.Resultados" %>

<!-- Sección de contenido para el encabezado de la página -->

<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">

<title>Resultados</title>

</asp:Content>

<!-- Sección de contenido principal de la página -->

<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">

<!-- Contenedor centrado para el título y el ganador -->

<div class="text-center">

<h2>Resultados de las Elecciones</h2>

<h3>El ganador es: <span id="lblGanador" runat="server"></span></h3>

</div>

<!-- Contenedor para la tabla de resultados -->

<div class="table-container">

<!-- GridView para mostrar los resultados de las elecciones -->

<asp:GridView ID="gvResultados" runat="server" CssClass="tabla-resultados" AutoGenerateColumns="False">

<Columns>

<!-- Definición de las columnas del GridView -->

<asp:BoundField DataField="Cedula" HeaderText="Cédula" />

<asp:BoundField DataField="Nombre" HeaderText="Nombre" />

<asp:BoundField DataField="Apellido1" HeaderText="Apellido1" />

<asp:BoundField DataField="Apellido2" HeaderText="Apellido2" />

<asp:BoundField DataField="Direccion" HeaderText="Dirección" />

<asp:BoundField DataField="Partido" HeaderText="Partido" />

<asp:BoundField DataField="CantidadVotos" HeaderText="Cantidad de Votos" />

<asp:BoundField DataField="Porcentaje" HeaderText="Porcentaje (%)" />

</Columns>

</asp:GridView>

</div>

</asp:Content>

**Resultado.apx.cs**

using System;

using System.Data.SqlClient;

using System.Web.Configuration;

namespace Sistema\_Votaciones

{

public partial class Resultados : System.Web.UI.Page

{

// Método que se ejecuta cuando la página se carga

protected void Page\_Load(object sender, EventArgs e)

{

// Verifica si la página se carga por primera vez y no es una recarga (postback)

if (!IsPostBack)

{

// Carga los resultados de las elecciones

CargarResultados();

}

}

// Método para cargar los resultados de las elecciones

private void CargarResultados()

{

// Obtiene la cadena de conexión de la configuración web

string connectionString = WebConfigurationManager.ConnectionStrings["BDVotacionesConnectionString"].ConnectionString;

// Usa la conexión con la base de datos

using (SqlConnection con = new SqlConnection(connectionString))

{

// Consulta SQL para obtener los resultados de las elecciones

string query = @"

SELECT C.Cedula, C.Nombre, C.Apellido1, C.Apellido2, C.Direccion, P.Nombre AS Partido, R.CantidadVotos, FORMAT(R.Porcentaje, 'N2') AS Porcentaje

FROM Resultados R

JOIN Candidatos C ON R.IDCandidato = C.Id

JOIN Partidos P ON C.IDPartido = P.Id

ORDER BY R.Porcentaje DESC";

// Crea el comando SQL con la consulta

using (SqlCommand cmd = new SqlCommand(query, con))

{

// Abrir la conexión

con.Open();

// Ejecuta la consulta y obtiene los resultados

SqlDataReader reader = cmd.ExecuteReader();

// Asignar los resultados al GridView para mostrarlos en la página

gvResultados.DataSource = reader;

gvResultados.DataBind();

// Cierra el reader después de usarlo

reader.Close();

}

// Consulta SQL para obtener el ganador de las elecciones

string winnerQuery = @"

SELECT TOP 1 C.Nombre, C.Apellido1, C.Apellido2, P.Nombre AS Partido

FROM Resultados R

JOIN Candidatos C ON R.IDCandidato = C.Id

JOIN Partidos P ON C.IDPartido = P.Id

ORDER BY R.Porcentaje DESC";

// Crea el comando SQL para obtener el ganador

using (SqlCommand winnerCmd = new SqlCommand(winnerQuery, con))

{

// Ejecuta la consulta para obtener el ganador

SqlDataReader winnerReader = winnerCmd.ExecuteReader();

// Verifica si hay resultados y asignar el nombre del ganador al label

if (winnerReader.Read())

{

string ganador = $"{winnerReader["Nombre"]} {winnerReader["Apellido1"]} {winnerReader["Apellido2"]} del partido {winnerReader["Partido"]}";

lblGanador.InnerText = ganador;

}

// Cerrar el reader después de usarlo

winnerReader.Close();

}

}

}

}

}

**Site.Master**

<%@ Master Language="C#" AutoEventWireup="true" CodeBehind="Site.master.cs" Inherits="Sistema\_Votaciones.Site" %>

<!-- Etiqueta HTML que define el idioma de la página -->

<html lang="en">

<head runat="server">

<!-- Metaetiqueta para definir la codificación de caracteres -->

<meta charset="utf-8" />

<title></title>

<!-- Enlace al archivo CSS para los estilos de la página -->

<link href="~/Content/Site.css" rel="stylesheet" />

<!-- Placeholder para contenido adicional en el head específico de cada página -->

<asp:ContentPlaceHolder ID="HeadContent" runat="server">

</asp:ContentPlaceHolder>

</head>

<body>

<!-- Formulario principal de la página -->

<form id="form1" runat="server">

<!-- Menú de navegación vertical -->

<div class="menu-vertical">

<ul>

<!-- Enlaces a las diferentes páginas del sitio -->

<li><a href="Default.aspx">Inicio</a></li>

<li><a href="IngresarCandidatos.aspx">Ingresar Candidatos</a></li>

<li><a href="Votaciones.aspx">Votaciones</a></li>

<li><a href="Resultados.aspx">Resultados</a></li>

</ul>

</div>

<!-- Placeholder para el contenido principal específico de cada página -->

<asp:ContentPlaceHolder ID="MainContent" runat="server" />

</form>

</body>

</html>

**Votaciones.aspx**

<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true" CodeBehind="Votaciones.aspx.cs" Inherits="Sistema\_Votaciones.Votaciones" %>

<!-- Contenido específico para la sección de Head -->

<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">

<title>Registrar Votaciones</title>

<!-- Incluyendo jQuery y Inputmask para el manejo de máscaras de entrada -->

<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/inputmask/5.0.6/inputmask.min.js"></script>

<script type="text/javascript">

// Script para aplicar máscaras de entrada a los campos de fecha de nacimiento

$(document).ready(function() {

var selector = $("#<%= txtFechaNacimiento.ClientID %>");

Inputmask("9999-99-99", { alias: "yyyy-mm-dd", placeholder: "YYYY-MM-DD" }).mask(selector);

Inputmask("99-99-9999", { alias: "dd-mm-yyyy", placeholder: "DD-MM-YYYY" }).mask(selector);

});

</script>

</asp:Content>

<!-- Contenido específico para la sección principal -->

<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">

<!-- Contenedor principal de la página -->

<div class="container">

<h2 class="text-center">Registrar Votaciones</h2>

<!-- Formulario para registrar votaciones -->

<div class="form-container">

<!-- Campo para el ID del votante -->

<div class="form-group">

<label for="IDVotante">ID Votante:</label>

<asp:TextBox ID="txtIDVotante" runat="server" CssClass="form-control" placeholder="Ingrese el ID Votante"></asp:TextBox>

</div>

<!-- Campo para el nombre del votante -->

<div class="form-group">

<label for="Nombre">Nombre:</label>

<asp:TextBox ID="txtNombre" runat="server" CssClass="form-control" placeholder="Ingrese el Nombre"></asp:TextBox>

</div>

<!-- Campo para el primer apellido del votante -->

<div class="form-group">

<label for="Apellido1">Apellido1:</label>

<asp:TextBox ID="txtApellido1" runat="server" CssClass="form-control" placeholder="Ingrese el Primer Apellido"></asp:TextBox>

</div>

<!-- Campo para el segundo apellido del votante -->

<div class="form-group">

<label for="Apellido2">Apellido2:</label>

<asp:TextBox ID="txtApellido2" runat="server" CssClass="form-control" placeholder="Ingrese el Segundo Apellido"></asp:TextBox>

</div>

<!-- Campo para la fecha de nacimiento del votante -->

<div class="form-group">

<label for="FechaNacimiento">Fecha de Nacimiento:</label>

<asp:TextBox ID="txtFechaNacimiento" runat="server" CssClass="form-control" placeholder="YYYY-MM-DD or DD-MM-YYYY"></asp:TextBox>

<asp:CustomValidator ID="cvFechaNacimiento" runat="server" ControlToValidate="txtFechaNacimiento" ErrorMessage="Fecha inválida. Use el formato YYYY-MM-DD o DD-MM-YYYY." OnServerValidate="cvFechaNacimiento\_ServerValidate"></asp:CustomValidator>

</div>

<!-- Campo para el correo electrónico del votante -->

<div class="form-group">

<label for="Email">Email:</label>

<asp:TextBox ID="txtEmail" runat="server" CssClass="form-control" placeholder="Ingrese el Email"></asp:TextBox>

</div>

<!-- Campo para la dirección del votante -->

<div class="form-group">

<label for="Direccion">Dirección:</label>

<asp:TextBox ID="txtDireccion" runat="server" CssClass="form-control" placeholder="Ingrese la Dirección"></asp:TextBox>

</div>

<!-- Campo para seleccionar el candidato a votar -->

<div class="form-group">

<label for="Candidato">Candidato:</label>

<asp:DropDownList ID="ddlCandidatos" runat="server" CssClass="form-control"></asp:DropDownList>

</div>

<!-- Botón para registrar el voto -->

<div class="form-group text-center">

<asp:Button ID="btnRegistrarVoto" runat="server" CssClass="btn" Text="Registrar Voto" OnClick="btnRegistrarVoto\_Click" />

</div>

</div>

</div>

</asp:Content>

**Votaciones.aspx**

using System;

using System.Data.SqlClient;

using System.Configuration;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace Sistema\_Votaciones

{

public partial class Votaciones : System.Web.UI.Page

{

// Método que se ejecuta al cargar la página

protected void Page\_Load(object sender, EventArgs e)

{

// Verifica si la página no es una recarga para cargar los candidatos

if (!IsPostBack)

{

CargarCandidatos();

}

}

// Método que se ejecuta al hacer clic en el botón de registrar voto

protected void btnRegistrarVoto\_Click(object sender, EventArgs e)

{

// Valida la fecha de nacimiento

DateTime fechaNacimiento;

if (!DateTime.TryParse(txtFechaNacimiento.Text, out fechaNacimiento))

{

cvFechaNacimiento.ErrorMessage = "Fecha inválida. Use el formato YYYY-MM-DD o DD-MM-YYYY.";

cvFechaNacimiento.IsValid = false;

return;

}

// Verifica si el votante es mayor de 18 años

if (fechaNacimiento.AddYears(18) > DateTime.Now)

{

cvFechaNacimiento.ErrorMessage = "Debe ser mayor de 18 años.";

cvFechaNacimiento.IsValid = false;

return;

}

// Obtiene la cadena de conexión de la configuración

string connectionString = ConfigurationManager.ConnectionStrings["BDVotacionesConnectionString"].ConnectionString;

using (SqlConnection con = new SqlConnection(connectionString))

{

// Consulta SQL para insertar el voto

string query = "INSERT INTO Votos (IDCandidato, IDVotante) VALUES (@IDCandidato, @IDVotante)";

using (SqlCommand cmd = new SqlCommand(query, con))

{

// Agrega parámetros a la consulta

cmd.Parameters.AddWithValue("@IDCandidato", ddlCandidatos.SelectedValue);

cmd.Parameters.AddWithValue("@IDVotante", txtIDVotante.Text);

// Abre la conexión y ejecutar la consulta

con.Open();

cmd.ExecuteNonQuery();

}

}

// Limpia los campos del formulario

LimpiarCampos();

}

// Método para cargar los candidatos en el DropDownList

private void CargarCandidatos()

{

// Obtiene la cadena de conexión de la configuración

string connectionString = ConfigurationManager.ConnectionStrings["BDVotacionesConnectionString"].ConnectionString;

using (SqlConnection con = new SqlConnection(connectionString))

{

// Consulta SQL para obtener los candidatos con su partido

string query = "SELECT C.Id, C.Nombre + ' ' + C.Apellido1 + ' ' + C.Apellido2 + ' (' + P.Nombre + ')' AS NombreCompleto FROM Candidatos C JOIN Partidos P ON C.IDPartido = P.Id";

using (SqlCommand cmd = new SqlCommand(query, con))

{

// Abre la conexión y ejecutar la consulta

con.Open();

SqlDataReader reader = cmd.ExecuteReader();

// Establece la fuente de datos del DropDownList

ddlCandidatos.DataSource = reader;

ddlCandidatos.DataTextField = "NombreCompleto";

ddlCandidatos.DataValueField = "Id";

ddlCandidatos.DataBind();

}

}

}

// Método para limpiar los campos del formulario

private void LimpiarCampos()

{

txtIDVotante.Text = "";

txtNombre.Text = "";

txtApellido1.Text = "";

txtApellido2.Text = "";

txtFechaNacimiento.Text = "";

txtEmail.Text = "";

txtDireccion.Text = "";

ddlCandidatos.SelectedIndex = 0;

}

// Validador personalizado para la fecha de nacimiento

protected void cvFechaNacimiento\_ServerValidate(object source, ServerValidateEventArgs args)

{

DateTime fechaNacimiento;

// Valida el formato de la fecha de nacimiento

if (!DateTime.TryParseExact(args.Value, new[] { "yyyy-MM-dd", "dd-MM-yyyy" }, null, System.Globalization.DateTimeStyles.None, out fechaNacimiento))

{

args.IsValid = false;

}

else

{

// Verifica si el votante es mayor de 18 años

args.IsValid = fechaNacimiento.AddYears(18) <= DateTime.Now;

}

}

}

}

**Web.config**

**<**configuration>

<!-- Sección para definir las cadenas de conexión a la base de datos -->

<connectionStrings>

<!-- Cadena de conexión a la base de datos 'BDVotaciones' usando SQL Server -->

<add name="BDVotacionesConnectionString"

connectionString="Data Source=DESKTOP-TQUE363\SQLEXPRESS;Initial Catalog=BDVotaciones;Integrated Security=True"

providerName="System.Data.SqlClient" />

</connectionStrings>

**Demostración de la Funcionalidad de la Pagina Web**

**Página de Inicio**

**Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente**

La página de inicio del Sistema de Votaciones presenta al usuario su propósito principal: facilitar el proceso electoral. El diseño de esta página es atractivo y fácil de usar; tiene una interfaz clara y una imagen que enfatiza el mensaje de participación democrática.

**Encabezado de Bienvenida:** Este encabezado recibe a los visitantes y explica el propósito de la página.

**Imagen de Votación:** La imagen destaca la importancia de cada voto y refuerza visualmente el concepto de participación en el proceso electoral.

**Página de Ingresar Candidatos:**

**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente**

Los administradores del sistema pueden agregar nuevos candidatos a la base de datos electoral a través de la página "Ingresar Candidatos". Esta página tiene campos fáciles de entender y una disposición lógica que guía al usuario a través del proceso de ingreso de datos.

**Propósito y Uso**

Facilitar el ingreso de datos de candidatos: esta página está diseñada para facilitar el ingreso de datos de candidatos y garantizar que toda la información relevante se capture de manera organizada y efectiva.

Validación de Datos: Los campos del formulario se validan para garantizar que los datos estén intactos y se minimicen los errores de entrada.

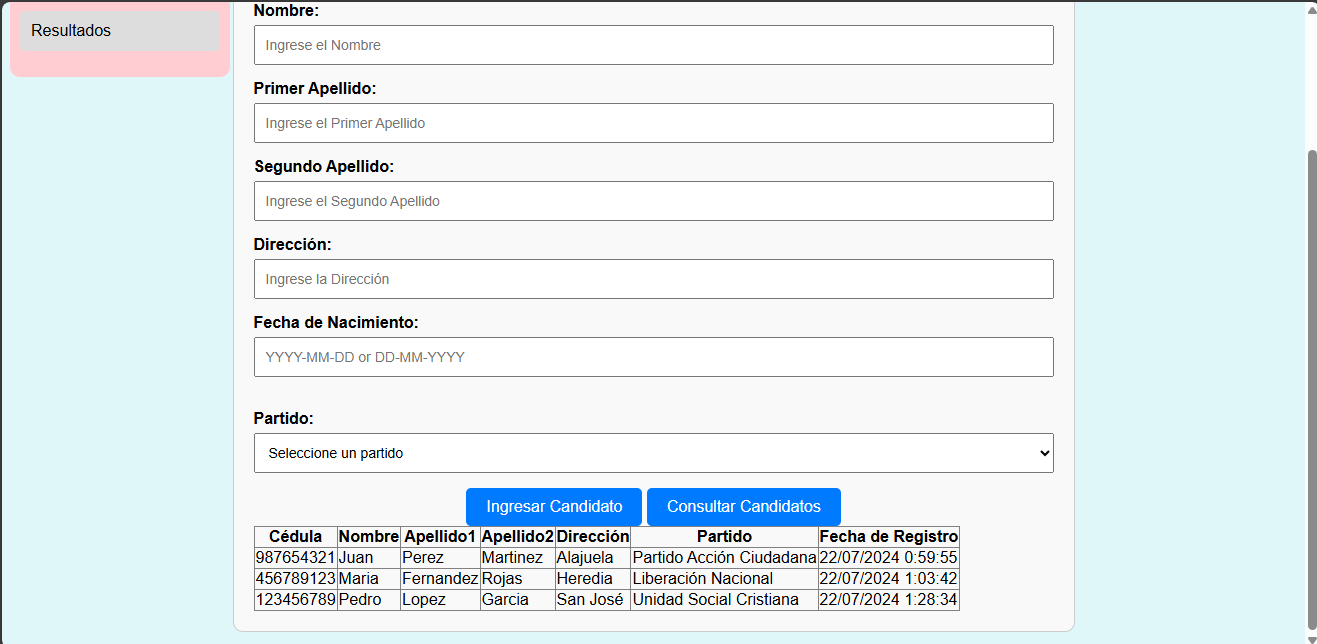
Interfaz Amigable: Los placeholders y la disposición clara de los campos de entrada hacen que la página sea fácil de entender y fácil de usar incluso para usuarios sin experiencia técnica.

Ejemplo de como se visualiza al agregar un candidato:

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Se ingresan los datos y podemos consultar los candidatos ingresados:



**Pagina Votaciones**

**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente**

Formulario de Registro de Votos: En el centro de la página, se encuentra un formulario para registrar los votos. Este formulario tiene varios campos que los usuarios deben completar:

**ID Votante:** Campo para ingresar el número de identificación del votante.

**Nombre:** Campo para ingresar el nombre del votante.

**Primer Apellido:** Campo para ingresar el primer apellido del votante.

**Segundo Apellido:** Campo para ingresar el segundo apellido del votante.

**Fecha de Nacimiento:** Campo para ingresar la fecha de nacimiento del votante, con soporte para dos formatos (YYYY-MM-DD y DD-MM-YYYY).

**Email:** Campo para ingresar el correo electrónico del votante.

**Dirección:** Campo para ingresar la dirección del votante.

**Candidato:** Lista desplegable para seleccionar el candidato por el que se va a votar.

El usuario completa los campos del formulario con su información personal y selecciona un candidato.

Al hacer clic en "Registrar Voto", se valida la información.

Si la validación es exitosa, el voto se guarda en la base de datos.

El formulario se limpia para permitir el ingreso de nuevos datos.

**Página de Resultados:**

**Interfaz de usuario gráfica

Descripción generada automáticamente**

La página de resultados está diseñada para mostrar los resultados de las elecciones de una manera clara y concisa.

La tabla de resultados muestra la información detallada de cada candidato participante en las elecciones. Los campos que se presentan en la tabla son:

Cédula: Número de identificación del candidato.

Nombre: Nombre del candidato.

Apellido1: Primer apellido del candidato.

Apellido2: Segundo apellido del candidato.

Dirección: Dirección del candidato.

Partido: Nombre del partido al que pertenece el candidato.

Cantidad de Votos: Número total de votos recibidos por el candidato.

Porcentaje: Porcentaje de votos obtenidos por el candidato respecto al total de votos válidos.

**Cálculo y Visualización del Ganador:** El ganador se determina automáticamente y se muestra en el subtítulo, destacando su nombre completo y el partido al que pertenece. En este caso, Juan Perez Martinez del partido Partido Acción Ciudadana ha ganado con el 50% de los votos.

La página se carga y muestra los resultados automáticamente al acceder, sin necesidad de recargar manualmente los datos. Esto se realiza a través de consultas SQL que extraen la información de la base de datos y la presentan en la interfaz de usuario.